Anadolu Üniversitesi Bilim ve Teknoloji Dergisi A- Uygulamalı Bilimler ve Mühendislik
Anadolu University Journal of Science and Technology A- Applied Sciences and Engineering

ANADOLU UNIVERSITY

2017 - Volume: 18 Number: 1
Page: 218 - 224
DOI: 10.18038/aubtda.270276
Received: 18 November 2017 Revised: 01 December 2016 Accepted: 26 January 2017

THE IMPACT OF TEXT REPRESENTATION AND PREPROCESSING ON AUTHOR IDENTIFICATION

Muhammet Yasin PAK [1,*], Serkan GÜNAL [1]

[1] Department of Computerl Engineering, Faculty of Engineering, Anadolu University, Eskişehir, Turkiye

ABSTRACT

Author identification, one of the popular topics in text classification and natural language processing, basically aims to determine the author of a given text through various analyses. In the literature, different text representation approaches and use of preprocessing steps are considered for author identification problem. This paper aims to comprehensively examine the impact of text representation and preprocessing steps on author identification specifically for Turkish language. For this purpose, the contributions of all possible combinations of different text representation approaches, namely unigram and bigram, together with the preprocessing tasks, including stemming and stop-word removal, to the performance of author identification are investigated. For the experimental evaluation, a brand new dataset is constituted. Also, two different classification algorithms, namely Multinomial Naive Bayes and Sequential Minimal Optimization, are employed. The results of the experimental analysis reveal that using bigram features alone should be avoided. Besides, it is shown that stop-words should be kept inside the text while stemming can be preferred depending on the classification algorithm so that higher performance can be achieved for author identification.

Keywords: Author identification, Text classification, Text preprocessing, Text representation

1. INTRODUCTION

The amount of text on the internet has been increasing dramatically day by day. As a result of this increase, author analysis over text has gained great importance. Author analysis is one of the hot research topics at the intersection of text classification and natural language processing. Author analysis fundamentally aims to retrieve particular information about the author of a given text by performing certain analyses on that text. Author analysis mainly focus on four problems namely author identification, author verification, similarity detection and author characterization [1]. Author identification detects the author of the given text from a set of authors whereas author verification aims to verify whether the text is written by a specific author or not. Similarity detection finds similarity between two texts and author characterization aims to extract personal information about the author such as age, gender and educational level.

Considering the author identification for Turkish text, the efforts on this area are quite limited unlike English or other widely used languages. In [2], as one of few examples in Turkish, the dataset, which contains 360 columns belonging to 18 authors, is constituted. In that study, 22 of style markers are determined and employed as the features for author identification. In another work [3], a system is developed to classify Turkish text based on the authors, genres and author genders. Character *n*-grams are used as the features. Various feature vectors are used to identify authors of Turkish texts in another work [4]. Ten different feature vectors are arranged using the features such as frequent words, 2- and 3-character *n*-grams, linguistic and statistical features. According to the results of the experiments, it is observed that character *n*-gram features provide better results than other features when used alone. The highest scores are achieved using Naive Bayesian and Support Vector Machine classifiers. In [5], the author of a given column is identified using the dataset consisting of 35 columns for each of 18 authors in total. In order to determine the authorship attribution performance when employing the

*Corresponding Author: mypak@anadolu.edu.tr

homogeneous and heterogeneous documents, and 3 datasets with different sizes are constituted. If the average results for 3 datasets are considered, the best result is obtained when character bigram and trigram are used as the features and feature selection is carried out. In another work [6], a new method that use term frequency and document frequency for weighting is proposed. The number of features is reduced to the number of classes so that classification performance is improved. In [7], some linguistic studies are carried out to determine important characteristics of Turkish by using a large scale Turkish corpus.

Also, author identification problem is handled using word *n*-gram for 16 authors. In [8], 50 columns belonging to 17 authors are used. Chi-square algorithm is applied for feature selection and 17 features out of 20 are selected.

While the abovementioned works propose various feature extraction, feature selection and classification approaches to improve the performance of author identification in Turkish language, the impact of text representation and preprocessing steps have not been extensively examined so far. This paper aims to fulfill this task. Specifically, the contributions of all possible combinations of different text representation approaches, namely unigram and bigram, together with the preprocessing tasks, including stemming and stop-word removal, to the performance of author identification are investigated. For the experimental evaluation, a brand new dataset, which consists of 6000 columns of 60 different authors published in a national newspaper in Turkiye, is constituted. During the investigation, two different classification algorithms, namely Multinomial Naive Bayes (MNB) and Sequential Minimal Optimization (SMO), are employed. The results of the experimental analysis reveal that stop-word removal is not absolutely necessary as a preprocessing task in author identification; however, stemming can be preferred in some cases of text representation.

The remainder of the paper is organized as follows: Section 2 briefly explains text representation approaches for text classification. Section 3 explains the preprocessing methods including stop-word removal and stemming. Section 4 describes the experimental work and the related results in detail. Finally, some concluding remarks are given in Section 5.

2. TEXT REPRESENTATION

As mentioned before, the representation method of text have direct impact on the success of author identification. In the literature, various features have been used to represent text in author identification problem [9]. Average sentence length, word count, punctuation count, content-specific words are just some examples to those features. *n*-gram methods are also commonly used in author identification. There are two approaches used for *n*-gram methods in general: character and word. While each character or word can be a feature, *n* sequential characters or words can constitute a feature as well. In other words, for character *n*-gram approach, a contiguous sequence of *n* letters from a given sequence of text is generated. On the other hand, in word *n*-gram approach, an *n*-gram is a contiguous sequence of *n* words from a given text after the text is tokenized into the words. Word *n*-gram approach is often referred to the bag-of-words model, probably the most common approach in text classification [10-13].

In this work, the features are obtained by assigning *n* as 1 and 2 so that unigram and bigram features are attained, respectively. Additionally, a third feature set was constituted by the combination of unigram and bigram features. The features were weighted using term frequency - inversed document frequency (TF-IDF) approach [14].

3. PREPROCESSING METHODS

Just like any other text classification problem, the preprocessing is also one of the fundamental stages in author identification. This stage usually includes the tasks such as tokenization, stop-word removal,

lowercase conversion and stemming. Specifically, stop-word removal and stemming are considered in this work. Stop-words may be identified as the words that are commonly encountered in text regardless of a particular topic. For this reason, they are, most of the time, assumed to be uninformative. However, there exists several efforts, which reveals this assumption is not always true [15]. As one can easily realize, stop-words are specific to the language. Table 1 lists the common stop-words for Turkish language [16].

Another critical preprocessing step in text classification is stemming, which aims to obtain stem or root forms of the words in the given text. Since the derived words and their root forms are semantically similar, stemming step is usually carried out before term frequencies are computed. As expected, stemming algorithms depend on the language that is being studied. For Turkish language, there are different stemming approaches such as Zemberek [17] and fixed prefix method [16]. Due to its simplicity and computational efficiency, the fixed-prefix algorithm is preferred in this work to carry out stemming task. The value of 5 was chosen in the algorithm as used in many works previously [12].

Table 1. Sample stop-words for Turkish language

acaba, altı, ama, ancak, arada, aslında, ayrıca, bazı, belki, ben, beri, bile, bin, bir, birçok, birkaç, biz, böyle, bu, burada, çok, çünkü, da, daha, dahi, de, defa, değil, diğer, diye, dolayı, eden, eğer, en, gibi, göre, halen, hangi, hatta, hem, henüz, hep, hepsi, her, herhangi, hiç, için, ile, ilgili, ise, işte, itibaren, kadar, karşın, kendi, kez, ki, kim, milyon, mu, mı, nasıl, ne, neden, nerede, nereye, niye, niçin, o, olmak, olsa, onu, oysa, öyle, pek, rağmen, sadece, sanki, sen, siz, şey, şöyle, şu, tarafından, tüm, üzere, var, ve, veya, ya, yani, yapılan, yerine, yine, yirmi, yoksa, yüz, zaten

4. EXPERIMENTAL WORK

In this section, the content of the dataset, the feature selection and the classification algorithms utilized within the experiments are first described. Then, the experimental procedure and the respective results are presented.

4.1. Dataset

For this work, the columns belonging to 60 different authors from the web site of a national newspaper published in Turkiye were collected and a brand new dataset was constituted. The dataset contains 6000 columns in total with 100 columns for each author. The topics of the columns are mainly on politics, sports, economics, health as well as the other popular issues. For fair evaluation, the numbers of columns for each author were kept identical. 10-fold cross validation was used during the evaluation.

4.2. Feature Selection

Feature selection, one of the critical stages in text classification, evaluate a given set of features to obtain a more discriminative lower-dimensional subset. In this way, both processing time and classification performance of text classification can be enhanced. In the literature, though there are numerous approaches to feature selection for text classification, those methods mainly fall into two categories, namely filters and wrappers [18]. The filters evaluate features independently from a classification algorithm whereas the wrappers assess features by employing a particular classifier. In this work, simple but effective term frequency approach, which is a type of the filters, was preferred. Specifically, the terms with the term frequencies less than 10 were eliminated while the others are kept.

4.3. Classification Algorithms

Two classification algorithms, namely Multinomial Naive Bayes (MNB) and Sequential Minimal Optimization (SMO), were employed in this work.

MNB is one of the probabilistic models for classification which make the naïve Bayes assumption [19]. In multinomial model, a document is an ordered sequence of word event and this model captures word frequency information inside documents. The model assumes that the probability of each word event in a document is independent of context and position of the word within the document.

Sequential Minimal Optimization (SMO) is an implementation of Support Vector Machine (SVM) in Weka library [20]. SMO is a simple and efficient algorithm to solve SVM Quadratic Programming (QP) problem by breaking this large problem into a series of smallest possible QP problems without any extra matrix storage [21]. SMO can handle very large training sets since the amount of memory required for SMO is linear in the training set size.

4.4. Procedure and Results

The experiments were carried out by evaluating all possible combinations of different text representation approaches and preprocessing tasks. The text representation approaches include unigram (U), bigram (B) and their combination (U+B) whereas the preprocessing tasks are selected as stemming (ST) and stop-word removal (SR). The combinations of the preprocessing tasks are listed as (ST=OFF | SR=OFF), (ST=OFF | SR=ON), (ST=ON | SR=OFF) and (ST=ON | SR=ON) where ON and OFF keywords symbolically indicate that the related preprocessing task is enabled and disabled, respectively. It should also be noted that non-alphabetic character removal and lowercase conversion were applied during the experiments.

The results of the experimental analysis for two classification algorithms are illustrated in Figures 1 and 2. F-score values are given in those figures for each combination of the preprocessing tasks together with the text representation approaches. Considering the entire analysis for two classifiers, the highest F-score (84.20%) was achieved by MNB classifier when (U+B) text representation was used and both preprocessing steps were disabled (ST=OFF | SR=OFF). On the other hand, the lowest F-score (46.40%) was attained by SMO classifier in case that bigram representation alone was used, stemming was disabled and stop-word removal was enabled (ST=OFF | SR=ON). When the two classification algorithms were compared, it was observed that MNB classifier outperformed SMO classifier most of the time.

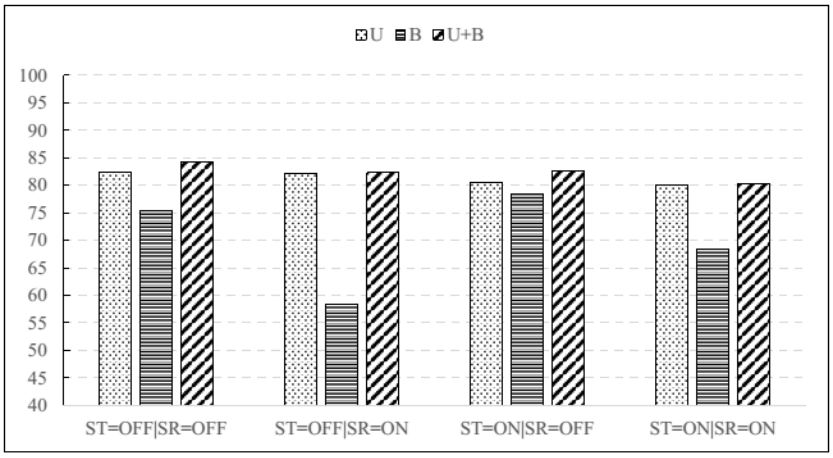

Figure 1. The experimental results for MNB classifier (F-score %)

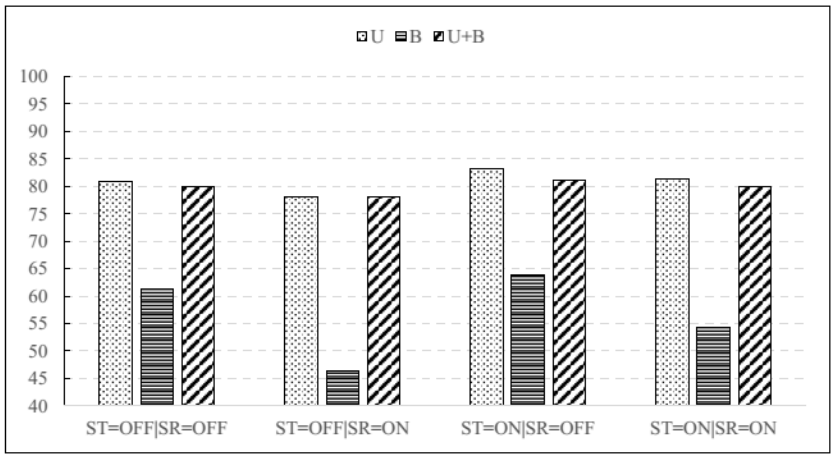

Figure 2. The experimental results for SMO classifier (F-score %)

From the perspective of text representation, (U) and (U+B) representations offered similar classification performance whereas (B) representation alone was able to achieve significantly lower performance than those of the other two representation approaches. This statement was valid for all combinations of the preprocessing tasks and the classification algorithms.

From the preprocessing point of view, disabling the two preprocessing tasks (ST=OFF | SR=OFF) offered slightly better performance than those of the other combinations of the preprocessing steps when MNB classifier was preferred. This statement was valid for all text representation approaches except (B) alone. On the other hand, enabling just the stemming task (ST=ON | SR=OFF) made it possible to achieve the highest F-score for SMO classification algorithm. This statement was valid for all three text representation approaches.

Considering the text representation approaches and the preprocessing tasks together, it can be clearly stated that when bigram representation was used alone, the classification performance dropped substantially almost all the time. Furthermore, the performance got even worse for all cases when stop-word removal is enabled (SR=ON) in addition to (B) text representation.

In case of MNB classifier, when the combinations of the preprocessing tasks are compared, it can be observed that use of stemming and stop-word removal steps decreased the success of classification for unigram; however, stop-word removal had a little effect. In case of using (B) representation, different results were obtained in comparison to (U) representation. While using stemming provided better performance for (B) representation, stop-word removal significantly decreased the success. If the results are considered in terms of text representation methods, the feature sets including (U) provided better performance with respect to the ones containing (B). Finally, the highest performance was achieved as 84.20% when (U+B) representation was used and both preprocessing steps were disabled (ST=OFF | SR=OFF).

In case of SMO classification algorithm, it was observed that the use of stemming step enhanced the accuracy for (U) representation whereas stop-word removal had a negative effect. In case of using (B) representation, the results were similar to the ones achieved by MNB classifier. The feature sets including (U) provided better outcome with respect to using only (B) when the text representation methods were taken into consideration likewise MNB classifier. The best performance was achieved as 83.20% when (U) representation was preferred and only stemming step was enabled (ST=ON | SR=OFF).

5. CONCLUSIONS

In this work, the impact of text representation and preprocessing tasks on author identification is examined specifically for Turkish language. All possible combinations of the text representation approaches and the fundamental preprocessing tasks are considered. For the experimental evaluation, two different classification algorithms are employed. Besides, a brand new dataset is constituted to be used for the experimental work.

According to the results of the experimental evaluation, it can be stated that bigram features should not be used alone. Using either unigram or the combination of unigram and bigram features would guarantee to achieve higher performance for author identification in Turkish. Since the highest classification performance for each classifier is attained when stop-word removal is disabled, keeping stop-words inside the text would also help to obtain better performance even if different classification algorithm is used. However, stemming step might be necessary depending on the classifier utilized for author identification.

Analysis of the contributions of the other text representation approaches and preprocessing methods to the success of author identification task in both Turkish and other languages remain as important future works.

REFERENCES

[1] Aslanturk O. Turkish authorship analysis with an incremental and adaptive model. MSc, Hacettepe University, Ankara, Turkey, 2014.

[2] Diri B, Amasyali MF. Automatic author detection for Turkish texts. In: ICANN/ICONIP 2003 The Joint International Conference on Artificial Neural Networks and International Conference on Neural Information Processing; 26-29 June 2003; Istanbul, Turkey. pp. 138-141.

[3] Amasyali MF, Diri B. Automatic Turkish text categorization in terms of author, genre and gender. In: NLDB 11th International Conference on Applications of Natural Language to Information Systems; 31 May- 2 June 2006; Klagenfurt, Austria. pp. 221-226.

[4] Amasyali MF, Diri B, Turkoglu F. Farklı özellik vektörleri ile Türkçe dokümanların yazarlarının belirlenmesi. In: TAINN The 15th Turkish Symposium on Artificial Intelligence and Neural Networks; 21-24 June 2006; Mugla, Turkey.

[5] Turkoglu F, Diri B, Amasyali MF. Author attribution of Turkish texts by feature mining. In: ICIC 2007 The 3rd International Conference on Intelligent Computing; 21-24 August 2007; Qingdao, China. pp. 1086–1093.

[6] Kaban Z, Diri B. Genre and author detection in Turkish texts using artificial immune recognition systems. In: IEEE 2008 The 16th Signal Processing, Communication and Applications Conference; 20-22 April 2008; Aydin, Turkey. pp. 1-4.

[7] Orucu F. Turkish Language Characteristics and Author Identification. MSc, Dokuz Eylul University, İzmir, Turkey, 2009.

[8] Bay Y, Celebi E. Feature Selection for Enhanced Author Identification of Turkish Text. In: ISCIS 2015 The 30th International Symposium on Computer and Information Sciences; 23-25 September 2015; London, UK. pp. 371-379.

[9] Stamatatos E. A survey of modern authorship attribution methods. Journal of the American Society for Information Science and Technology 2009; 60: 538-556.

[10] Joachims T. A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization. In: ICML 1997 The 14th International Conference on Machine Learning; 8-12 July 1997; San Francisco, USA. pp. 143-151.

[11] Gunal S. Hybrid feature selection for text classification. Turkish Journal of Electrical Engineering & Computer Sciences 2012; 20: 1296-1311.

[12] Uysal AK, Gunal S, Ergin S, Sora Gunal E. The impact of feature extraction and selection on SMS spam filtering. Elektronika ir Elektrotechnika 2013; 19: 67-72.

[13] Pak MY, Gunal S. Sentiment classification based on domain prediction, Elektronika ir Elektrotechnika 2016; 22: 96-99.

[14] Manning CD, Raghavan P, Schtze H. Introduction to Information Retrieval. New York, USA: Cambridge University Press, 2008.

[15] Uysal AK, Gunal S. The impact of preprocessing on text classification. Information Processing & Management 2014; 50: 104-112.

[16] Can F, Kocberber S, Balcik E, Kaynak C, Ocalan HC, Vursavas OM. Information retrieval on Turkish texts. Journal of the American Society for Information Science and Technology 2008, 59: 407–421.

[17] Zemberek. <http://code.google.com/p/zemberek/> (Accessed December 2016).

[18] Gunal S, Edizkan R. Subspace based feature selection for pattern recognition. Information Sciences 2008; 178: 3716-3726.

[19] McCallum A, Nigam K. A comparison of event models for naïve Bayes text classification. AAAI-98 Workshop on Learning for Text Categorization 1998; 752: 41-48.

[20] Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH. The WEKA data mining software: an update. ACM SIGKDD Explorations Newsletter 2009; 11: 10-18.

[21] Platt JC. Fast training of support vector machines using sequential minimal optimization. In: Scholkopf B, Burges CJC, Smola AJ, editors. Advances in Kernel Methods. Cambridge, MA, USA: MIT Press, 1999. pp. 185-208.